

## Claims

We claim:

- 5
1. A computer-implemented method for programmatically generating a new graphical program, comprising:
- executing a graphical program generation (GPG) program;
- the GPG program receiving information, wherein the information specifies functionality of the new graphical program;
- the GPG program programmatically generating the new graphical program in
- 10 response to said information specifying the functionality of the new graphical program, wherein the new graphical program implements the specified functionality.
2. The method of claim 1, wherein said programmatically generating the new graphical program creates the new graphical program without any user input specifying
- 15 the new graphical program during said creating.
3. The method of claim 1, wherein the new graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the new graphical program.
- 20
4. The method of claim 1, wherein the new graphical program comprises a block diagram portion comprising a plurality of interconnected nodes and a user interface portion;
- wherein said programmatically generating the new graphical program includes
- 25 generating the block diagram portion and the user interface portion.
5. The method of claim 1, wherein said programmatically generating the new graphical program comprises:

*[Signature]*

interconnecting the plurality of graphical program objects in the new graphical m;

5

10

15

20

25

Conley, Rose & Tayon, P.C.

[illegible]

11. The method of claim 1,  
wherein the information received by the GPG program specifies a state diagram;  
wherein the GPG program is operable to generate a new graphical program that  
implements the specified state diagram.

12. The method of claim 1,  
wherein the information received by the GPG program specifies a prototype;  
wherein the GPG program is operable to generate a new graphical program that  
10 implements the specified prototype.

13. The method of claim 1,  
wherein the information received by the GPG program specifies a test executive  
sequence;  
15 wherein the GPG program is operable to generate a new graphical program that  
implements the specified test executive sequence.

14. The method of claim 1,  
wherein said GPG program receiving information comprises the GPG program  
20 receiving user input specifying desired functionality of the new graphical program;  
wherein the GPG program is operable to generate a new graphical program that  
implements the specified desired functionality.

15. The method of claim 14,  
25 wherein the GPG program comprises a graphical programming development environment application.

16. The method of claim 1,

Sub  
Part

wherein the information received by the GPG program specifies an instrumentation function;

wherein the GPG program is operable to generate a new graphical program that implements the specified instrumentation function.

5

17. The method of claim 16,

wherein the instrumentation function comprises one or more of:

a test and measurement function; or

an industrial automation function.

10

18. The method of claim 1,

wherein the information received by the GPG program comprises information regarding an existing program having program functionality;

wherein the GPG program is operable to generate a new graphical program that implements at least a portion of the program functionality of the existing program.

15

19. The method of claim 18,

wherein the existing program is a graphical program.

20

20. The method of claim 1,

wherein the GPG program is operable to generate a plurality of new graphical programs, depending on the received information.

25

21. The method of claim 1,

wherein the new graphical program generated by the GPG program has program functionality;

wherein the GPG program is operable to determine at least a portion of the program functionality independently of the received information.



*July 2002*

28 The method of claim 27,  
wherein the client portion of the GPG program executes in a first computer  
system;

5 wherein the server portion of the GPG program executes in a second computer  
system;

wherein the first computer system is connected to the second computer system.

29. The method of claim 1, further comprising:  
10 executing the new graphical program;  
wherein the new graphical program is operable to perform the specified  
functionality during execution.

30. The method of claim 1,  
15 wherein the new graphical program implements only a portion of the specified  
functionality.

31. The method of claim 1, wherein the new graphical program is a partial  
program, the method further comprising:  
20 adding additional graphical code to the new graphical program, in response to  
manual user input, in order to complete the new graphical program.

32. The method of claim 1,  
wherein said programmatically generating the new graphical program comprises  
25 including at least one graphical program object in the new graphical program.

33. The method of claim 32, wherein the new graphical program includes a  
block diagram, wherein the at least one graphical program object comprises a function  
node placed in the block diagram.

0000227-62054250

34. The method of claim 32, wherein the new graphical program includes a block diagram, wherein the at least one graphical program object comprises a programmatic structure placed in the block diagram.

35. The method of claim 32, wherein the new graphical program includes a user interface panel, wherein the at least one graphical program object comprises a user interface object placed in the user interface panel.

36. The method of claim 35, wherein the user interface object is a user interface input object placed in the user interface panel for performing one or more of: viewing input to the new graphical program; or providing input to the new graphical program.

37. The method of claim 35, wherein the user interface object is a user interface output object placed in the user interface panel for viewing output of the new graphical program.

38. The method of claim 35, wherein the new graphical program also includes a block diagram, wherein the user interface object is a user interface input object placed in the user interface panel for performing one or more of: viewing input to the block diagram; or providing input to the new graphical program.

39. The method of claim 35, wherein the new graphical program also includes a block diagram, wherein the user interface object is a user interface output object placed in the user interface panel for viewing output from the block diagram.

40. The method of claim 1, wherein said programmatically generating the new graphical program comprises:





*[Handwritten signature]*

45. The method of claim 42, wherein the GPG program further includes an invoke node; the method further comprising:

the invoke node invoking a method on the graphical program object in response to said executing the GPG program.

5

46. The method of claim 45, wherein the object creation node outputs a reference to the graphical program object;

wherein the invoke node receives as input the reference to the graphical program object;

10 wherein the invoke node invokes a method on the graphical program object  
specified by the reference to the graphical program object.

47. The method of claim 42, further comprising:

configuring the object creation node of the GPG program;

15 wherein said configuring comprises specifying a graphical program object class  
for the object creation node;

wherein the at least one graphical program object included in the new graphical program is of the graphical program object class.

20

48. The method of claim 1,

wherein the GPG program is a graphical program;

wherein the GPG program includes a graphical program creation node for programmatically creating the new graphical program.

25

49. A computer-implemented method for programmatically modifying an existing graphical program, comprising:

executing a GPG program;

*July 2003*  
the GPG program receiving information during program execution, wherein the information specifies functionality to add to the existing graphical program;

the GPG program programmatically modifying the existing graphical program in order to implement the specified functionality, in response to receiving the information.

5

50. The method of claim 49,

wherein said modifying the existing graphical program comprises adding graphical code to the existing graphical program.

10

51. The method of claim 49, wherein said programmatically modifying the existing graphical program modifies the existing graphical program without any user input specifying the modification to the existing graphical program during said modifying.

15

52. The method of claim 49,

wherein said receiving information during program execution comprises receiving user input specifying desired functionality to add to the existing graphical program.

20

53. A computer-implemented method for programmatically generating a new graphical program, comprising:

providing information specifying functionality of the new graphical program;

executing a graphical program generation (GPG) program;

the GPG program programmatically generating the new graphical program using

25

said information, wherein the new graphical program implements the specified functionality.

54. The method of claim 53, wherein said programmatically generating the new graphical program creates the new graphical program without any user input specifying the new graphical program during said creating.

55. The method of claim 53, wherein the new graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the new graphical program.

56. A memory medium for programmatically generating a new graphical program, the memory medium comprising program instructions executable to:

receive information, wherein the information specifies functionality of the new graphical program;

programmatically generate the new graphical program in response to said information specifying the functionality of the new graphical program, wherein the new graphical program implements the specified functionality.

57. The memory medium of claim 56, wherein said programmatically generating the new graphical program creates the new graphical program without any user input specifying the new graphical program during said creating.

58. The memory medium of claim 56, wherein the new graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the new graphical program.

59. The memory medium of claim 56, wherein the new graphical program comprises a block diagram portion comprising a plurality of interconnected nodes and a user interface portion;

wherein said programmatically generating the new graphical program includes generating the block diagram portion and the user interface portion.

60. A system for programmatically generating a new graphical program, the system comprising:

a processor coupled to a memory, wherein the memory stores a graphical program generation (GPG) program;

wherein the processor is operable to execute the GPG program in order to:

receive information specifying functionality of the new graphical program; and  
programmatically generate the new graphical program in response to said  
information specifying the functionality of the new graphical program, wherein the new  
graphical program implements the specified functionality.

61. The system of claim 60, wherein said programmatically generating the new graphical program creates the new graphical program without any user input specifying the new graphical program during said creating.

62. The system of claim 60, wherein the new graphical program comprises a plurality of interconnected nodes that visually indicate functionality of the new graphical program.

63. The system of claim 60, wherein the new graphical program comprises a block diagram portion comprising a plurality of interconnected nodes and a user interface portion;

wherein said programmatically generating the new graphical program includes generating the block diagram portion and the user interface portion.

094531-1000

creating a plurality of graphical program objects in the new graphical program;  
and

wherein the interconnected plurality of graphical program objects comprise at least a portion of the new graphical program.

000001094500